

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА**

**СОЗДАНИЕ WEB ПЛАТФОРМЫ CITY DASHBOARD**

[**Вступление**](#_heading=h.gjdgxs) **4**

[**Список сокращений**](#_heading=h.rqywqj8w5fqw) **6**

[**Основание для разработки платформы**](#_heading=h.cjtvqobomcj) **6**

[**Полное наименование работ**](#_heading=h.30j0zll) **7**

[**Цели и задачи проекта:**](#_heading=h.1fob9te) **7**

[**Этапы разработки и передача заказчику**](#_heading=h.er59olm63r07) **8**

[**Показатели достижения целей**](#_heading=h.n9hdt0s7mmce) **9**

[**Краткие требования к Системе**](#_heading=h.3znysh7) **9**

[**Характеристика объекта разработки**](#_heading=h.2et92p0) **10**

[**Сбор данных**](#_heading=h.w5gft3ej6rc1) **10**

[Хранение данных](#_heading=h.2s8eyo1) **12**

[Визуализация данных](#_heading=h.uhoujr5qasw1) **15**

[Аналитическая инфографика](#_heading=h.qkqoiwaok74) 20

[Тепловая карта](#_heading=h.qpopcr4cvvcq) 21

[Примеры визуализации взятые из открытых источников для сравнения](#_heading=h.i1r180ps6y38) 22

[Требования к панелям](#_heading=h.sf85f4shyi1a) 26

[Виды диаграмм](#_heading=h.bop495upakq4) 27

[Интеграция с социальными сетями](#_heading=h.dv29yh3sy3rx) 28

[**Дизайн системы**](#_heading=h.lfconli02qs0) **28**

[Пример UI KIT:](#_heading=h.26m432j3nca) 29

[Базовые требования](#_heading=h.fhx5qdlbiurq) 32

[Требования по операционным системам](#_heading=h.8nhcpslp1o6j) 33

[Требования по браузерам и платформам (кроссбраузерность)](#_heading=h.4td5ql216kpt) 33

[Требования к размерам](#_heading=h.jhw368h9mqyk) 34

[Требования к HTML коду](#_heading=h.3ve4xvgeg5g4) 34

[Требования к мобильной оптимизации](#_heading=h.5pa5whfyqrmk) 37

[**Управление модулем пользователей**](#_heading=h.bjvl71dfeyj0) **39**

[Авторизация пользователей](#_heading=h.wts6l6ltlbkx) 39

[Управления пользователями, ролями и личным кабинетом](#_heading=h.x4i0vqm0342s) 39

[**Требования к надежности**](#_heading=h.e9rxe1hd6vo9) **41**

[Перечень аварийных ситуаций](#_heading=h.kbypkrbenz8e) 42

[Требования к надежности технических средств и программного обеспечения](#_heading=h.d7c7sg7x5ss7) 42

[**Требования безопасности**](#_heading=h.37dosgruaqjx) **43**

[Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы](#_heading=h.cg4fe4ncr92a) 43

[Требования по стандартизации и унификации](#_heading=h.upuuug4v6lpw) 44

[Архитектура](#_heading=h.3tj1ncai8lyv) 45

[Диагностика](#_heading=h.bzmovgko0xui) 45

[Перспективы развития, модернизации системы](#_heading=h.2gnq2hgzou5p) 46

[Дополнительные требования](#_heading=h.hx0kfvjsbvzv) 46

[**Требования к платформе**](#_heading=h.o86sqii4x9w3) **46**

[Требования к виртуализации:](#_heading=h.ab63e2ejw4rw) 47

[Требования к серверному обеспечению со стороны Заказчика](#_heading=h.ji4mpcqand3z) 47

[**Требования к документированию**](#_heading=h.a2m55eqvcgrh) **48**

[**Ссылки**](#_heading=h.bawwdeyj2e3s) **48**

# Вступление

Информационное поле – это уже давно не поле, а целый океан. Зачастую разобраться в нем невероятно трудно. Чтобы помочь разобраться в данных, собрать их на одном экране и постоянно обновлять их, существуют Dashboard – интерактивная и динамическая визуализация.

Dashboard – панели анализа и мониторинга, визуализирующие данные в реальном времени. Визуализации на панели анализа и мониторинга появляются из отчетов, а каждый отчет создается на основе набора данных. Панель анализа и мониторинга обеспечивает объединенное представление локальных и облачных данных независимо от того, где они расположены. Фактически панель мониторинга можно рассматривать как средство представления базовых отчетов и наборов данных. Участие человека сведено к минимуму, так как все данные обновляются автоматически в режиме реального времени.

Концепция данного проекта заключается в создании интерактивной платформы для визуализации информации в аналитические инфографики и тепловые карты.

Интерактивная платформа должна визуализировать пользователю максимально детализированные данные в интуитивно **понятном** и доступном для понимания формате, с отражением аналитической информации с ассоциативными связями, которая должна заинтересовать пользователя провести разбор данных и даже способствовать к завлечению пользователя реагировать на полученные аналитические умозаключения.

Проект Civic City Dashboard – аналитическая, виртуальная платформа, которая отражает индикаторы и данные, которые позволяют пользователю понять, насколько эффективно работают муниципальные службы, каковы последствия принятых решений муниципалитетом, а также иметь общее представление о паттерне города. Основная цель наличия подобного инструмента создание доказательной базы дл принятия полиси документов, решений для развития города. Данный инструмент позволяет выстроить конструктивную практику гражданского участия и повышение подотчетности и прозрачности муниципальных служб и Мэрии городов данном проекте, вклад гражданского общества важен для сбора и распространения информации, соответственно портал City Dashboard должен включать в себя онлайн формы (опрос, анкета) для запроса информации и рекомендаций относительно определенных проблем и задач с последующим поощрением.

Информационная визуализация предлагает средства быстрого распространения информации и повышения его доступности между заинтересованными сторонами. Использование визуализации выступает в качестве промежуточной стадии при преобразовании данных в информацию. Средства визуализации используют визуальные и пространственные навыки человека, используя интерактивные визуальные представления данных. Визуализация может помочь принятию решений, помогая созданию пользовательской точной ментальной модели, которая может использовать когнитивные навыки [12-15]. Исследования в области когнитивной способности показывает, что люди могут обрабатывать больше информации представлены в графическом виде, чем в тексте [16 , 17]. Кейм [9] Говорится, что основные преимущества, предоставляемые визуальными в том, что они выступают в качестве кадра или временного хранения для познавательных процессов

человека. La Valle [18] Выделены визуализации как полезный инструмент для получения представления о больших и сложных наборах данных. Продольные наборы данных, полученные из других баз данных, как правило, многомерные, сложные, с различной степенью детализации [17], И, таким образом, может представлять подходящие наборы данных для использования средств визуализации. Datasets - наборы данных произведенное на из баз данных, как правило, доступно только в форматах, которые не позволяют возможности для быстрого извлечения и анализа информации

Например, Cheng [19] Разработал интерфейс - веб-интерфейс для обмена данными наблюдения гриппа в Гонконге. Дизайн основан на веб-сайтах в области общественного здравоохранения и руководящих принципов дизайна приборной панели. Ченг подчеркивает необходимость разработки рамок со стандартными наборами данных для включения нескольких источников и визуализировать различные заболевания. Sopan и др [24] Построили карту здоровья сообщества для обмена данными в области общественного здравоохранения среди различных пользователей, например, политиков, журналистов и общественности.

Приборная панель сосредоточена на возможность сравнения различных наборов данных и показателей, в том числе пространственной визуализации данных.

Разрабатываемая платформа должна обеспечить потребности в формировании подобных панелей, виджетов, аналитических обзоров с возможностью фильтрации пространства, времени и в комбинированном виде.

# Список сокращений

AJAX Асинхронный JavaScript и XML

API Применение программного интерфейса

BI Бизнес-аналитика

BSD Семейство разрешительных лицензий на свободное ПО

CSS Каскадируемые таблицы стилей

CSV Значения, разделенные запятыми

GPL General Public License

HCI Взаимодействие человека с компьютером

HTML Язык гипертекстовой разметки

IT Информационные технологии

JS JavaScript

KPI Ключевой показатель

MIT Бесплатная лицензия на программное обеспечение, происходящее из Массачусетского технологического института

NoSQL Ссылка на не реляционные базы данных

SD Среднеквадратичное отклонение

SQL Structured Query Language

SUS Юзабилити Scale System

W3C World Wide Web Consortium

# Основание для разработки платформы

Основанием проекта является необходимость предоставления гражданской общественности г.Бишкек Open Data ресурса с визуализацией данных в виде инфографиков и тепловой карты.

Опираясь на аналитические данные платформы гражданское общество должна принимать меры и влиять на улучшение инфраструктуры и жизни г.Бишкек.

# Полное наименование работ

Создание информационной платформы City Dashboard для сбора, хранения и визуализации данных с предоставлением открытого доступа для общественности – (далее – Система).

# Цели и задачи проекта:

1. Создание платформы для сбора, хранения и визуализации данных;
2. Создание возможностей сбора данных с различных источников:
   1. [API для взаимодействия с различными системами и БД;](#_heading=h.tyjcwt)
   2. [Сервис для парсинга официальных сайтов с открытыми данными](#_heading=h.3dy6vkm)
   3. [Создание форм для ручного ввода данных;](#_heading=h.1t3h5sf)
   4. [Стандартизация загружаемых отчётов в формате XLS и CSV;](#_heading=h.4d34og8)
3. [Хранение данных, получаемых с различных источников;](#_heading=h.2s8eyo1)
   1. [Модули для хранения данных и их состав полей, вывод данных, не прошедших стандартизацию для ручной корректировки;](#_heading=h.17dp8vu)
   2. [Настройка связей между данными](#_heading=h.3rdcrjn)
   3. [Автоматическая запись истории изменения данных.](#_heading=h.26in1rg)
4. Создание инструментов для визуализации данных:
   1. Графическая инфографика;
   2. Тепловая карта;
   3. Табличные данные.
5. Создание веб страниц для публичного доступа к Открытым данным:
   1. Инфографика;
   2. Тепловая карта;
   3. возможность проведения мониторинга и/или анализа определенной проблемы;
   4. SMM.
6. Управление пользователями

# Этапы разработки и передача заказчику

|  |  |  |
| --- | --- | --- |
| № | Наименование работ | Сроки реализации |
| 1 | Согласование перечня используемых платформ и библиотек.  Создание Дизайнов макета системы (по согласованию с заказчиком) | 15 дней |
| 2 | Разработка и демонстрация проекта системы (по согласованию с заказчиком) | 4 месяца |
| 3 | Подключение к базам открытых данных, службы 103, ГУОБДД | 15 дней |
| 4 | Настройка сервера, тест и запуск системы (по согласованию с заказчиком) | 7 дней |
| 5 | Исправление погрешностей и внедрение в эксплуатацию административной системы, системы управления контентом, личного кабинета и (по согласованию с заказчиком) | 15 дней |
| 6 | Акт приема - передачи кода\* системы  (по согласованию с заказчиком). Передача технической документации описания кода и развертывания системы | 7 дней |
| ВСЕГО | | 6 месяцев |

\*Передача исходного кода с правом на редактирование, использование и тиражирования.

# Показатели достижения целей

Для достижения этих целей, интерфейс панели должен будет соответствовать

следующим критериям проекта:

* Обеспечить монитор ключевых показателей эффективности (KPI).
* Включить пространственный опрос наборов данных в отношении ключевых индикаторов для выявления тенденций.
* Разрешить переход от глобального к локальному видению в отношении этих показателей.
* Детализация до локальных областей карты, для изучения наборов данных более подробно.

# Краткие требования к Системе

Требования к функционированию системы. Разрабатываемая Система должна учитывать следующие основные принципы:

1. Веб ориентированная платформа;
2. Стандартизированный API для подключения веб сервисов и DataSetов;
3. Удобный интерфейс для ввода данных в электронном формате (заполнение утвержденных форм, опросников, анкет) с возможностью прикрепления необходимых документов;
4. Модуль для стандартизации загружаемых данных в формате XLS, CSV;
5. Интерфейс для настройки системы;
6. Модули для хранения данных со следующими под модулями:
7. Визуализация данных для проведения мониторинга и анализа, с использованием ассоциативных данных;

# Характеристика объекта разработки

Объектом разработки является платформа для:

* сбора,
* хранения
* визуализации данных
  + в инфографиках
  + тепловых картах

Система должна иметь интерфейс на 3х языках – английском, русском и кыргызском языках. Платформа должна иметь возможность языкового переключения без потери страницы местонахождения. Фронт энд системы и Административная часть должна иметь интерфейс дополнения и правки переводов.

В рамках данного проекта планируется обеспечить разработку следующих модулей:

Веб-ориентированная система, которая представляет собой клиент-серверное приложение, в котором клиентскую часть реализует браузер, осуществляющий диалог с пользователем и отображение информации, а серверную часть – веб-сервер и сервер приложений, которые реализуют основную логику системы.

Платформа состоит из двух взаимосвязанных систем - система управление контентом и система визуализации контента.

# Сбор данных

**Стандартизированный API для подключения веб сервисов и DataSetов**

Необходимо создать веб сервисы для приема данных в различных форматах - API сервис, для удобного подключения с базами данных различных организаций.

***Требования к API системы***

* Надёжность (за счёт отсутствия необходимости сохранять информацию о состоянии клиента, которая может быть утеряна);
* Производительность (за счёт использования кэша);
* Масштабируемость;
* Прозрачность системы взаимодействия (особенно необходимая для приложений обслуживания [сети](https://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C));
* Простота [интерфейсов](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81);
* Портативность компонентов;
* Лёгкость внесения изменений;

**Сервис для парсинга официальных сайтов с открытыми данными**

Инструменты web scraping (парсинг) разработаны для извлечения, сбора любой открытой информации с веб-сайтов. Эти ресурсы нужны тогда, когда необходимо быстро получить и сохранить в структурированном виде любые данные из интернета. Парсинг сайтов – это новый метод ввода данных, который не требует повторного ввода или копипастинга.

**Создание форм для ручного ввода данных**

Формы предназначены для ввода данных в систему путем ручного набора с возможностью прикрепления данных и файлов. Формами могут быть:

* Опросники;
* Анкеты;
* Формы ввода данных для модулей не имеющих DataSet.

**Функция Drag & Drop для загружаемых отчётов в формате XLS и CSV**

Способ оперирования элементами интерфейса в интерфейсах пользователя (как графическим, так и текстовым, где элементы GUI реализованы при помощи псевдографики) при помощи манипулятора «мышь» или сенсорного экрана.

Способ реализуется путём «захвата» (нажатием и удержанием главной (первой, чаще левой) кнопки мыши) отображаемого на экране компьютера объекта, программно доступного для подобной операции, и перемещении его в другое место (для изменения расположения) либо «бросания» его на другой элемент (для вызова соответствующего, предусмотренного программой, действия). По отношению к окнам (также способным к перемещению подобным способом) данный термин обычно не употребляется.

Базовыми действиями и самыми простыми примерами drag-and-drop действий являются: перемещение объекта, перемещение объекта из панели в панель, хотя в современных операционных системах drag-and-drop получил широкое применение и является одним из главных способов взаимодействия с компьютером в графическом интерфейсе пользователя.

Необходимо автоматически определять данные, если выявилась ошибка, то выводить пользователю для корректировки.

# Хранение данных

**Типы данных**

Наборы данных могут содержать следующие типы данных:

* Текст
* Цифры, деньги и проценты;
* Даты / время (iso-8601);
* Логические значения;
* Адреса электронной почты;
* URL-адреса;
* Таблицы;
* Геолокационные метки;
* Списки.

В первую очередь все данные должны хранится в распределённой базе данных с кластеризацией.

Созданные поля в системе должны иметь валидацию на тип вводимых данных. В случае возникновения ошибки при вводе данных, система должна выводить уведомление пользователю со ссылкой на модуль - ID запись. Данные ошибки должны фиксироваться в соответствующем модуля для последующего формирования отчетов и принятия мер по недопущению ошибок.

**Модули для хранения данных и их состав полей**

Для хранения данных в системе необходимо создавать Соответствующие модули и поля.

* + Модуль Документы – загружаемые документы в формате XLS, CSV:
    - Наименование;
    - Дата создания;
    - Папка хранения документа;
    - Кем создано
    - Номер документа
    - Тип файла
    - Размер файла
    - Активный
    - Связи с другими модулями системы
  + Модуль для DataSet – службы 103 (Центра экстренной медицины):
    - ID записи
    - Название полей
    - Даты создания записи
    - и т.д.

Состав полей зависит от согласованности с руководством службы 103

* + Модуль для DataSet – ГУОБДД:
    - ID записи
    - Название полей
    - Даты создания записи
    - и т.д.

Состав полей зависит от согласованности с руководством ГУОБДД

* + Модуль для создания форм Опросников и Анкет:
    - ID записи
    - Название полей
    - Даты создания записи

По согласованию с заказчиком

* + Модуль GPS точек с названиями улиц, мкр., организации и т.п.;
    - Название
    - Долгота
    - Широта
    - Дата создания
    - Кем создана
  + Модуль Комментарии
    - ID записи
    - Дата создания
    - Связь с модулем
    - Связь с пользователем;

**Настройка связей между данными**

Необходимо создавать связи как между модулями, так и между полями. Связи необходимы для отображения в инфографике и тепловой карте ассоциативные связи.

Связь должна быть отражена как ссылка и интуитивно понятна для пользователей системы. Также связи должны быть удобными для использования в верстке.

**Автоматическая запись истории изменения данных**

Для хронологии изменения данных автоматическая запись даты изменения данных в модуле обязательно. Хронологию данных можно группировать со следующей периодичностью:

* Ежечасно
* Ежедневно
* Еженедельно
* Ежемесячно
* Ежеквартально
* Полугодие
* Ежегодно.

**Функции обработки данных**

Для сбора, хранения и соответственно обработки данных в системе должны разработаны следующие функции:

* Все необходимые функции для сбора данных
* Все функции для хранения данных
* Все функции для формирования аналитических и статистических данных в инфографику и тепловую карту, формирование отчетов
* обеспечение безопасности информации;
* одновременное выполнение операций различными пользователями (не менее 500 пользователей одновременно);

# Визуализация данных

На сегодняшний день количество платформ интерактивных досок и виджетов с открытым исходным кодом превалирует над количеством платных. Многие из них это всего лишь отрывок из библиотек JavaScript, которые обеспечивают построения графиков с интерактивными элементами (например, реакция мыши над событиями). Ограничение на JavaScript связана с его доминирующей доли рынка для клиентской стороны языков, используемых для веб-страниц (90%).

Ниже приведены платформы и библиотеки, рекомендуемые к использованию. Перед началом реализации исполнитель обязан предоставить перечень систем управления контентом, библиотек и технологий, планируемых в использовании в реализации проекта.

Таблица: Список приборной панели / виджетов обработки структур

|  |  |  |
| --- | --- | --- |
| **Наим-е** | **Характеристики** | **Лицензия** |
| [**Atlasboard**](http://atlasboard.bitbucket.org/) | **С открытым исходным кодом, модуль для Node.js** | **Apache** |
| **angular dashboard framework** | **С открытым исходным кодом, модуль для Angular.js** | **MIT** |
| **Dashing** | **С открытым исходным кодом, Ruby on Rails** | **MIT** |
| [**FnordMetric**](http://fnordmetric.io/) | **С открытым исходным кодом для построения графиков с помощью SQL** | **GPL** |
| [**Gridster.js**](http://gridster.net/) | **С открытым исходным кодом, JQuery плагин** | **MIT** |
| [**jDash**](http://www.jdash.net/) | **Для пользователей Asp.Net** | **Коммерческий** |
| [**jSlate**](http://jslate.com/) | **При условии, как обслуживание** | **Свободное** |
| [**Pyxley**](http://multithreaded.stitchfix.com/blog/2015/07/16/pyxley/) | **С открытым исходным кодом, для пользователей Python** | **MIT** |
| [**RazorFlow**](https://www.razorflow.com/) | **С открытым исходным кодом** | **Коммерческий** |
| [**Reportr**](http://www.reportr.io/) | **С открытым исходным кодом, построен с использованием Node.js** | **Apache** |
| **Shiny** | **С открытым исходным кодом, для пользователей R** | **GPL** |

Таблица: Список интерактивных графиков библиотек JavaScript

|  |  |  |
| --- | --- | --- |
| **Наим-е** | **Характеристики** | **Лицензия** |
| [**C3.js**](http://c3js.org/) | **С открытым исходным кодом, построен с использованием D3.js** | **MIT** |
| [**Chart.js**](http://www.chartjs.org/) | **Открытый источник** | **MIT** |
| [**D3.js**](http://d3js.org/) | **Открытый источник; Настраиваемая визуализация библиотеки BSD** |  |
| [**dc.js**](https://dc-js.github.io/dc.js/) | **С открытым исходным кодом** | **Apache** |
| [**gRaphael**](http://g.raphaeljs.com/) | **Построенная с****использованием Raphael** | **MIT** |
| [**JavaScript InfoVis**](http://philogb.github.io/jit/index.html) | **Открытый источник** | **MIT** |
| [**NVD3.js**](http://nvd3.org/) | **С открытым исходным кодом, построен с использованием D3.js** | **Apache** |
| **Проект EON** | [**С открытым**](http://www.pubnub.com/developers/eon/) **исходным кодом, для данных в режиме реального времени, используя встроенный C3.js** | **MIT** |

Все данные платформы в зависимости от назначенной роли пользователя могут быть доступны для проведения мониторинга и анализа данных. Необходима возможность настройки ассоциативных связей между информационными модулями.

1. Хранение данных необходима для последующей ее визуализации на страницах инфографики и тепловой карте. Создание ассоциативных связей между массивами данных для отображения в Инфографике и тепловой карте;
2. Визуализация данных в удобном виде для пользователя с возможностью выбора отображения данных (графики, диаграммы и т.п.) и ассоциативных связей.

Для визуализации данных требуется разработать дизайн системы со следующими функциями:

1. Инфографики и тепловые карты (не менее тематических 10 шаблонов):

* **Главное окно** и страницы ввода данных
* **Аналитическая инфографика** –визуализация данных для проведения аналитики с использованием инструментов и функционалов
* **Панель мониторинга** - панель данных в котором отображаются текущие online данные, получаемые с различных источников. Например: онлайн видео потоки, динамичные данные по заполняемым формам опросов, данные с различных датчиков и т.п.
* **Тепловая карта** – отображение данных на карте с использованием инструментов и функционалов
* **Администрирование системы** – панель для настройки системы

Для работы с Инфографикой и Тепловой картой необходимо реализовать следующие функции:

* **группировка**— способ объединения схожих данных (по какому-то общему признаку, например по первой букве слова или имени человека);
* **агрегация**(сумма, минимум, максимум, количество и т.д.) — способ отображения колонки фактов из исходной базы данных (например, уникальное количество посетителей сайта, или сумма расходов на продукты);
* **сортировка**— упорядочивание уже сгруппированных данных по заданному признаку (кроме алфавита, можно отсортировать фамилии менеджеров по их наибольшим продажам за месяц и т.п.);
* **фильтрация**— исключение данных по заданному признаку или сложной формуле;
* **вычисляемая колонка** — способ получения новых данных и знаний с использованием методов работы с датами, строками, математических функций (например отображение имени и фамилии, вычисление возраста согласно дате рождения и текущей дате);
* **топовые (лучшие) значения** — способ отобразить указанное количество максимальных или минимальных значений данной группировки (например, возраст трёх самых молодых сотрудников крупной компании, или пять менеджеров, обеспечивающих максимальные продажи);
* **виджеты**(таблицы, диаграммы, карты и т.п.) — собственно способ визуализации вышеуказанных понятий.
* **Временная шкала** – для анализа данных в разрезе времени
* **Виды графиков** - Изменение отображения данных в различные типы графиков и диаграмм на усмотрение пользователя
* **Переход** - Отображение данных с переходом с инфографики на тепловую карту и наоборот
* **Sharing** – поделиться в соц. сетях и мессенджерах
* Сопоставление со смежными данными по ассоциативным данным
* Выгрузка и распечатка данных

Административная панель – для настройки системы

Приоритетные функции административной панели:

* Менеджер модулей - управление модулями, их функциями, связями и полями.
* Менеджер ролей и профилей - управление доступами к данным, спискам, справочникам, возможностям публикации и модулям в целом
* Менеджер интеграции - управление вебсервисами, критериями взаимодействия, ограничениями объемов и временных параметров.

### Аналитическая инфографика

Инфографика — это подборка изображений и диаграмм с минимумом сопроводительного текста, позволяющая быстро понять суть освещаемой темы. Используются красочные и привлекательные графические объекты для быстрого и четкого донесения информации.

Инфографика должна используется для:

* быстрого раскрытия темы,
* объяснения сложных процессов,
* представления результатов исследований и опросных данных,
* сравнения и противопоставления различных вариантов чего-либо,
* повышения осведомленности о проблеме или представления идей.

Инфографика в данном проекте должна:

* Привлекательная и понятная тема;
* Иметь плавный, красивый, эффективный дизайн
* Использовать эмоциональные цвета
* иметь краткую, но содержательную информацию;
* должна быть динамичной – движущиеся (анимированные) изображения;
* Использовать шкалу времени
* Предлагать варианты отображения данных, в различных графиках, диаграмм и с переходом на тепловую карту и наоборот.

### Тепловая карта

Тепловая карта — это отображение данных на карте города Бишкек имеющих GPS координаты с сопроводительной информацией, позволяющая быстро сориентироваться по местоположению данных. Используются различные слои отображения карты города Бишкек.

Для удобства восприятия данных должны быть следующие виды отображения:

* Снимок спутника - цветная
* Схема – монотонная (Google Map, 2GIS, OSM)

Карта отображает следующий данные:

* Любые данные системы, которые имеют GPS точку или контур, со всплывающей информацией при наведении на объект.
* Административные участки:
  + Мэрии города Бишкек - границы города, районов, микрорайонов и т.п.;
  + Станции скорой помощи – границы обслуживания подстанций и всей обслуживаемой территории по городу Бишкек.
  + ГУВД г.Бишкек:
    - РОВД
    - ПОМ
    - ГОМ
  + Иные объекты по требованию клиента
* Базу данных адресов и организаций со смежными данными
* Данные с онлайн веб сервисов:
  + Погода
  + Курс валют
  + Пробки
  + Видеопотоки
  + Различные показатели: качество воздуха
  + И т.п.

Информация, отраженная на карте, должна иметь возможность перехода на инфографику и наоборот.

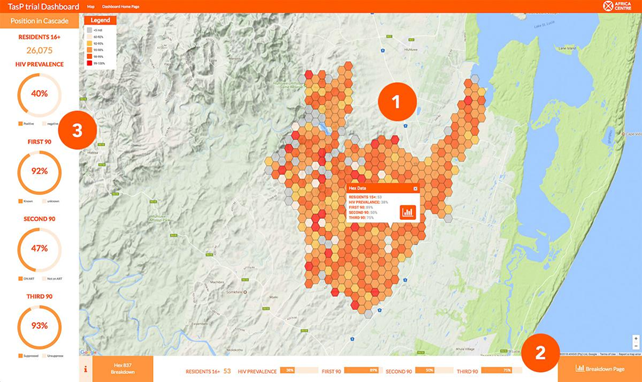
Инструменты, используемые на карте:

* Поиск
  + Организации
  + Адрес
  + Пересечения
* Задать область для отражения данных:
  + Круг
  + Квадрат
  + Произвольная форма
* Проложить маршрут от и до
* Информационные слои, с возможностью включения и отключения
* Линейка – для определения расстояние
* Полигональная линейка – для определения площади

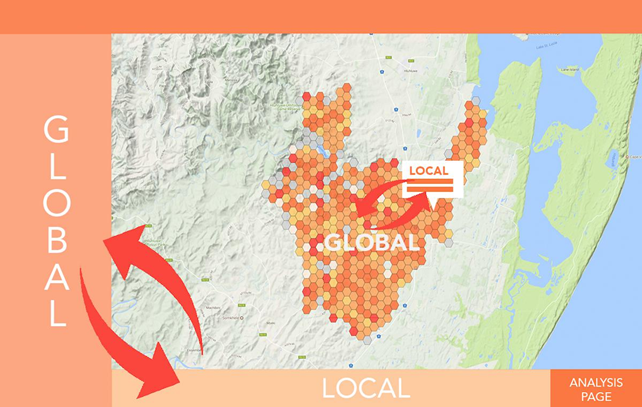
Все информационные данные, получаемые с различных источников, должны обязательно храниться в базе данных с указанием даты создания. Это необходимо для функционала «временная шкала», которая помогает пользователю проводить аналитику в разрезе времени.

### Примеры визуализации взятые из открытых источников для сравнения

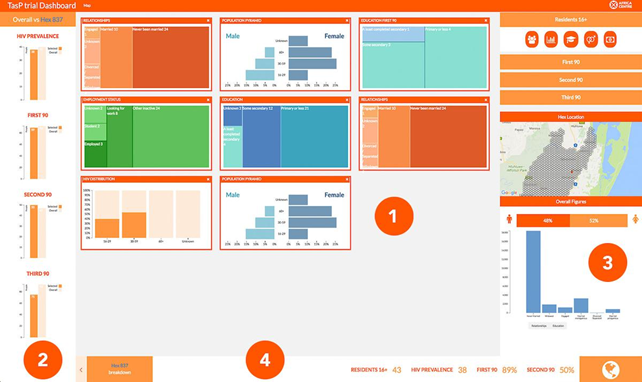
Данные обновляются в режиме реального времени. Пользователю предлагается управлять картой и изменить выбранные регионы с помощью мыши или прикосновением. Кнопка разбивка Page приводит к более детальному исследованию мер и связанных с ними факторов в пределах страницы анализа.



Точки сравнения. Это изображение подчеркивает точки взаимодействия между глобальными и локальными показателями, с указанием точек, в которых пользователь имеет возможность непосредственно сравнить локальные и глобальные меры ключевых показателей эффективности.



Анализ страницы. На этом рисунке показана страница анализа для локального региона, выбранного в рамках глобальной страницы карты. На странице выводится целый ряд потенциальных объясняющих признаков, которые могут быть измерены на основе ключевых показателей эффективности, которые помогают разрабатывать гипотезы для исследований в будущем. Пользователи имеют возможность добавлять и удалять визуализации с помощью интерактивных функций, а также менять временные рамки.



Использование уровней увеличения детализации от (а) ключевого показателя (б) контекста вокруг метрики с полной детализацией. Ниже интерфейс Aperture интернет-рекламы приборной панели Datran Media, который показывает эту модель:

ключевые метрики

контекст 

подробности

Необходимо реализовать визуальные подсказки и функциональные возможности, чтобы привлечь пользователей к вещам, которые наиболее важны. Некоторые механизмы, которые могут помочь в этом - позиционирование на странице, и правильное использование цветовых гамм и шрифтов**.**

### Требования к панелям

* Детализация: Возможность перехода из сводной метрики или в целях дополнительной детализации, что обеспечивает более контекст и / или прорыв информации.
* Фильтры: позволяет пользователям определить объем данных в панели управления. Фильтры могут быть глобальным или местного (для специфической диаграммы или метрики, или представления).
* Сравнение: Возможность видеть два или более подмножеств бок о бок данных. Линия диаграммы, например, может позволить пользователю просматривать два географических региона в виде отдельных линий.
* Оповещения: Информация “Highlight” на основе заранее определенных критериев. Оповещение может быть активировано, когда метрика выходит за пределы определенного порога.
* Экспорт / печать: Возможность пользователям извлекать информацию из панели. Экспорт в форматы, которые позволяют пользователям производить иные действия с данными - Excel и CSV, а не PDF.

### Виды диаграмм

В платформе должны быть реализованы возможности применения построения не менее 11 типов различных диаграмм, перечисленных ниже:

* Линейный график
* Гистограмма
* Пузырьковый график
* Лепестковая диаграмма
* Японские свечи
* Диаграмма Венна
* QR-код
* Карта
* Формула
* Граф
* Круговая диаграмма

**Приемка дизайна**

Оценка конструкции каркаса панелей должна быть проведена с помощью проведения опросов соответствия требования путем испытания панелей. Эти панели данных должны быть оценены путем оценки ее использования 5 пользователями предложенными и согласованными с клиентом. В платформе должны быть интегрированы данные из источников, предложенные клиентом, реализованные на заранее подготовленных темах, профессионально разработанных с наличием;

* настройки текста, диаграмм, цветов;
* карт, иллюстраций и таблиц;
* сохранение, публикация в социальных сетях или загрузка в виде JPG, PNG, PDF;
* интерактивные графики;

### Интеграция с социальными сетями

Платформа должна обеспечить интеграцию с социальными сетями, с возможностью публикации данных соц. сетей на виджетах, а также обратную публикацию данных в социальных сетях. Интеграции подлежать популярные следующие соц. сети (не менее 5):

* Facebook
* Twitter
* Youtube
* иные указанные клиентом.

Возможность поделиться ссылкой в (не менее в 5):

* Whatsapp
* Gmail
* Telegram
* иные указанные клиентом.

Кроме того, возможность получения HTML – кода для вставки на сайт (гиперссылки) и загрузка изображений. Необходимо логическое разделение панелей по типу представляемой информации: пространственный, хронологический, количественный или смешанные комбинации.

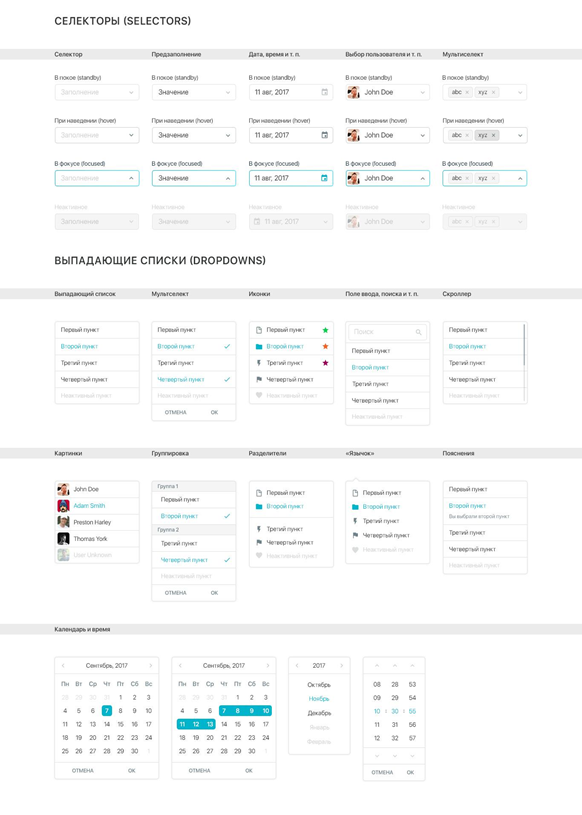
# Дизайн системы

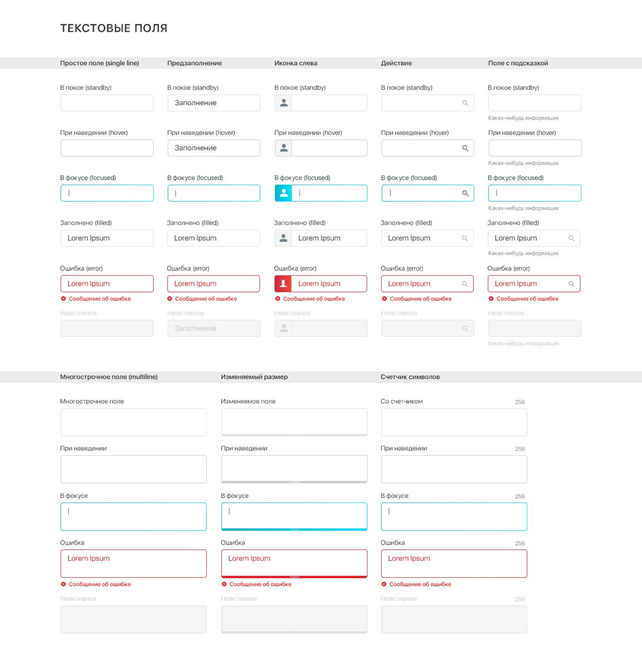
У системы должен быть удобный и интуитивно понятный интерфейс для каждого пользователя. Цвета и шрифты должны гармонировать с основными элементами.

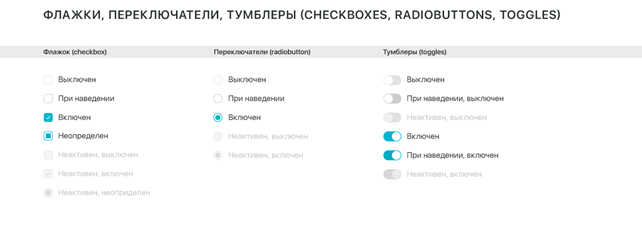
Все страницы системы должны быть оптимизированы для просмотра на всех типах мониторов, электронных мобильных устройств, включая смартфоны, планшеты, ноутбуки и смарт ТВ.

В каскадном стиле программирования дизайна – веб интерфейса необходимо нарисовать UI KIT (UI kit — это набор [элементов пользовательского интерфейса](http://fimushkin.com/blog/design/ui/).)

### Пример UI KIT:







### Базовые требования

Для резиновых макетов обязательно должна быть задана минимальная и максимальная ширина.

* Цвет фона для body задан всегда. Если конкретный цвет не указан в макете, используется белый (#fff). В случае если фон под контентом отличается от общего фона страницы, для body его нужно прописать в html-коде.
* Недопустимы грубые ошибки в разметке (ссылки сделаны не тегом <a>, абзацы должны быть абзацами, а не <br><br>, формы должны быть только внутри тега <form>).
* Запрещается использовать транслит в названиях классов, атрибутах и так далее.
* Запрещается строить сетку с помощью таблиц и позиционирования.
* Необходимо указать альтернативные варианты шрифта и тип семейства в конце перечисления.
* Все теги должны быть корректно закрыты.
* Должна быть прописана кодировка. В теге body не должно быть тега script.
* Подзаголовки h1, h2 и так далее в качестве элементов дизайна ни в коем случае не должны использоваться.
* h1 на странице должен быть один. h2, h3 и прочие не должны быть выше h1.
* Все стили должны быть вынесены в style.css. Там должны быть прописаны стили заголовков, списков, параграфов и прочих элементов.
* Для ссылок должны быть прописаны стили (цвет, подчеркивание) в файле CSS.
* При выборе между средствами javascript и чистыми HTML5/CSS3 безусловное предпочтение должно отдаваться последним. Если все-таки присутствует javascript, предпочтительно выносить его в отдельный файл. В целом код должен быть как можно короче.

Вёрстка проходит тест на переполнение контентом. Вёрстка не ломается:

* При добавлении в элементы большего количества текста;
* При использовании картинок с неподходящими размерами;
* Текст не должен выпадать из объектов;
* Переполнение контентными блоками не приводит к нарушению сетки.

### Требования по операционным системам

Серверная часть системы должна быть реализована на базе операционной системы с открытым исходным кодом. В обязанности исполнителя входит подготовка серверного программного обеспечения.

### Требования по браузерам и платформам (кроссбраузерность)

Верстка веб-сайта должна поддерживаться браузерами в следующих версиях на следующих платформах:

На персональных компьютерах и ноутбуках и мобильных устройствах:

* Internet Explorer - версия 10 и выше;
* Microsoft Edge - все версии;
* Google Chrome - версия 29 и выше;
* Yandex Browser - версия 13 и выше;
* Mozilla Firefox - версия 18 и выше;
* Opera - версия 15 и выше;
* Safari - 7.1 и выше.

Верстка полностью должна соответствовать дизайну. Скрипты работают в соответствии с ТЗ. Минимальные показатели отклонения от макета равны 3-5 пикселям.

### Требования к размерам

Блоки с изменяющимся содержимым подстраиваются в соответствии с дизайном при уменьшении/увеличении контента.

Страницы при масштабировании страницы в диапазоне 70 – 150 % в браузерах IE9, Chrome 15+, Opera 12+, Safari 5, FF15+ должны выглядеть так же, как и при 100%. Допускаются не большие погрешности, которые возникают из-за неправильных округлений координат браузеров.

### Требования к HTML коду

* Кодировка – utf-8
* Структурный, комментируемый код (обозначается начало/конец крупных блоков). Отбивка табами.
* [Семантическая разметка](http://www.xiper.net/learn/tegofenshuj/) на уровне грамотного использования тегов.
* Имена классов и идентификаторов – осмысленные, на усмотрение исполнителя.
* Классы служат для привязки оформления, идентификаторы - скриптов.

Требования к стилям и CSS коду

Структурный, отбивка табами.

* Комментариями обозначены начало/конец крупных модулей/блоков разметки.
* Допускается использование вендорных префиксов.
* Стили для IE 9+ вынесены в отдельные CSS.
* Для IE 9+ для реализации не поддерживаемых CSS свойство допустимо использование Javascript и expression.

*Порядок CSS-свойств*

Приблизительная схема расположения свойств:

{

Позиционирование Параметры блока Размеры

Таблицы / списки Свойства текста Шрифт

Цвет

}

Требования к Javascript коду

* Структурный, отбивка табами.
* Имена переменных осмысленные, на усмотрение исполнителя.
* Снабжен комментариями: описаны назначения
  + - методов/классов
    - функций
    - условий.
* Код должен быть без ошибок.
* Общие требования к наименованию классов:
  + - HTML разметка
    - Для разметки используются теги HTML5.
    - Значения всех атрибутов заключаются в двойные кавычки (").
    - Форматирование HTML иерархическое, отступ - один символ табуляции.
* Используется блочная верстка.
* В контентных фрагментах верстки используются теги <p>, <span>. Не допускается использование блочных элементов основной разметки в контентных фрагментах верстки.
* В качестве разделителя отдельных элементов имени класса или идентификатора равнозначно используются символы дефиса (-) или нижнего подчеркивания (\_).
* Использование символа дефиса предпочтительнее. Использование классов и идентификаторов равнозначно.
* Названия классов и id должны по смыслу соответствовать применению (например, header, menu, footer, news).
* Для именования классов и идентификаторов используются следующие префиксы:
  + - применяемые в фреймворках;
    - обозначающие разметку HTML;
    - обозначающие визуальное оформление элементов.

### Навигация

Пользовательский интерфейс системы должен обеспечивать наглядное, интуитивно понятное представление структуры веб-сайта, быстрый и логичный переход к разделам и страницам. Навигационные элементы должны обеспечивать однозначное понимание пользователем их смысла: ссылки на страницы должны быть снабжены заголовками, условные обозначения соответствовать общепринятым. Графические элементы навигации должны быть снабжены альтернативной подписью.

Система должна обеспечивать навигацию по всем доступным пользователю ресурсам и отображать соответствующую информацию. Для навигации должна использоваться система контент-меню. Меню должно представлять собой текстовый блок (список гиперссылок) в верхней части страницы.

Для разделов, содержащих подразделы, должно быть предусмотрено выпадающее подменю.

При выборе какого-либо из пунктов меню пользователем должна загружаться соответствующая ему информационная страница (новостная лента, форма обратной связи и пр.), а в блоке меню (или в основной части страницы в зависимости от утвержденного дизайна) открываться список подразделов выбранного раздела.

Открытая пользователем страница должна отображать навигационную цепочку (хлебные крошки, breadcrumbs)

### Требования к мобильной оптимизации

Система (все страницы) должна быть оптимизирована для всех мобильных устройств, включая смартфоны и планшеты.

Скорость загрузки мобильной версии АИС на тестировочной платформе Google Developers PageSpeed Insights должна быть не ниже 80.

Все содержимое сайта должно входить по ширине в экран устройства. Внизу не должно быть горизонтальной прокрутки, или справа - пустого пространства:

Размер шрифта в мобильной версии сайта должен отображаться корректно. Он не должен быть слишком крупным или мелким, а должен быть удобным для чтения и быть адаптированным под экран мобильного устройства.

Все интерактивные элементы - элементы, на которые можно кликнуть в мобильной версии веб-сайта также должны быть удобными для нажатия.

Для быстрого набора их с мобильных устройств, все номера телефонов должны иметь активные ссылки. Например, пользователь просто кликает по номеру телефона в АИС и сразу же переходит к звонку:

**Требования к SEO оптимизации веб-сайта**

Система в части публичного портала должна быть оптимизирована для поисковых систем Google и Yandex.

Меню сайта необходимо делать текстовым, т.е. элементы меню могут быть графическими, но текст в виде графики - реализация через CSS.

У каждого изображения на сайте должны быть прописаны alt и title. Если не вручную, то автоматически.

Не использовать в качестве навигационных элементов Flash.

Все JavaScript и стили надо выносить в отдельные файлы, чтобы не засорять код.

На сайте не должно быть ссылок на несуществующие документы или битые картинки.

Открытая пользователем страница должна отображать навигационную цепочку (хлебные крошки, breadcrumbs).

**Применение каскада**

Каскад стилей должен быть сведен к минимуму. Он допускается, в частности, в случае, когда необходимо оформить контент, редактируемый в WYSIWYG-редакторе; этот контент по возможности должен не содержать классов.

**Требования к изображениям**

Проходят базовую оптимизацию на уровне оптимизации для Веб в Photoshop.

Форматы используемых изображений:

- для логотипов: SVG, PNG;

- для визуального оформления: SVG, PNG;

- контентные изображения: JPEG, PNG.

Для контентных изображений используется размер, не превышающий по своей ширине ширину полосы основного контента веб-страницы.

Для галерей изображений используется препроцессор подготовки миниатюр изображений предустановленного размера, без потери качества относительно исходного изображения.

Для изображений в формате JPEG используется однородная степень сжатия, не менее 85%. При сжатии не должно возникать видимых дефектов характерных данному формату.

# Управление модулем пользователей

### *Авторизация пользователей*

Платформа должна иметь личный кабинет для участников проекта. Администратор системы создает пользователей или выдает роль зарегистрированным самостоятельно через социальные сети (Gmail, Facebook) на право внесения данных и создания собственных видов данных и панелей представления.

### *Управления пользователями, ролями и личным кабинетом*

В личном кабинете пользователи могут работать как по шаблону, так и с чистого листа. Все сохраненные работы отображаются в личном кабинете пользователя. В случае предоставления администратором системы пользователь получает возможность публикации данных в общем реестре. В личном кабинете необходимо реализовать возможность импорта данных в популярных машиночитабельных форматах csv, xml, dbf, json. Кроме того, обеспечить возможность интеграции данных путем API напрямую с источниками данных.

Все действия пользователей должны регистрироваться в журнале действий.

В административной панели администратору предоставляется возможность создание ролей с доступом как управления другими ролями, так и управлениями пользователями платформы. Функционал должен иметь возможность определения доступа к модулям, конкретным полям в виде доступа на создание, редактирование и просмотр.

# Требования к надежности

Система должна обеспечивать работоспособность круглые сутки ежедневно в течение календарного года за исключением периодов проведения плановых регламентных работ по обслуживанию и восстановления Системы после аварий.

Система должна предоставлять возможность проведения плановых регламентных работ по обслуживанию оборудования и программного обеспечения:

· обновление версий программного обеспечения;

· обновление баз данных;

· резервное копирование.

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

· при сбоях в системе электроснабжения аппаратной части, приводящих к перезагрузке операционной системы (ОС), восстановление программы должно происходить после перезапуска ОС и запуска исполняемого файла Системы;

· при ошибках в работе аппаратных средств (кроме носителей данных и программ) восстановление функции Системы возлагается на ОС;

· при ошибках, связанных с программным обеспечением (ОС и драйверы устройств), восстановление работоспособности возлагается на ОС.

Надежность подсистем должна характеризоваться следующими показателями:

· среднее время восстановления – не более 8 часов;

· среднее время наработки на отказ – не менее 7000 часов;

· назначенный срок службы – не менее 3 лет.

Платформа должна относиться к малообслуживаемым самовосстанавливаемым системам специального назначения многократного применения в круглосуточном режиме.

Состав и количественные значения показателей надежности для системы в целом или ее подсистем:

Надежность системы в целом, определяемая как способность выполнять заданные функции, сохраняя во времени значения установленных эксплуатационных показателей в заданных пределах при заданных условиях эксплуатации, должна оцениваться по каждой функции в отдельности.

Среднее время восстановления работоспособности программной части АИС не должно превышать период в 30 минут.

Гарантийный срок эксплуатации АИС должен составлять не менее 12 месяцев. Значения показателей:

• коэффициент готовности платформы должен быть не менее 99%;

• среднее время восстановления работоспособности программно-технического комплекса системы составлять не более 8 часов, при этом:

среднее время восстановления работоспособности системы в случае отказа или сбоя из-за алгоритмических ошибок прикладного или системного программного обеспечения, без устранения которых невозможно дальнейшее функционирование Системы — до 8 часов (без учета времени на устранение ошибок).

### Перечень аварийных ситуаций

Отказы внешних систем или сбои взаимодействия с внешними системами не должны приводить к аварийному отказу.

Неправильные действия непривилегированных пользователей не должны приводить к возникновению аварийной ситуации.

Плановая остановка не должна приводить к сбою в работе программного обеспечения или нарушениям данных.

В случае сбоя система должна обеспечивать возможность консистентного восстановления данных до состояния, позволяющего администратору возобновить или повторить прерванные операции.

При запуске и остановке системы все сопутствующие процессы должны проходить в автоматическом режиме.

### Требования к надежности технических средств и программного обеспечения

Система должна обеспечивать защиту от отказов одного или нескольких элементов/серверов системы без потери функционала.

Система должна обеспечивать горизонтальное масштабирование модулей (без зависимости одного от другого) с пропорциональным увеличением мощности поставляемого оборудования (выделения виртуализированных ресурсов).

# Требования безопасности

Уровень безопасности системы должно соответствовать действующим мировым стандартам безопасности с соблюдением всех правовых, нормативных, договорных нормативных документов.

Инфраструктура безопасности системы должна включать брандмауэр и усиленную политику безопасности на всех серверах системы. ПО Системы должно иметь включенные по умолчанию политики безопасности для всех модулей, если не предусмотрено иное.

Требования по безопасности распространяются на все подсистемы системы, включая приложения, СУБД, ОС, предназначенные для обработки (передачи, хранения) информации, недоступность, искажение или раскрытие которой может привести к нанесению ущерба самой системе, оператору или пользователю системы.

Сервисы, интерфейсы и протоколы, не имеющие отношения к функционированию системы и ее компонентов (включая сетевое и серверное оборудование, общесистемное и прикладное ПО — ОС, СУБД, приложения и т.д.) должны быть отключены.

Компоненты системы не должны содержать недостатков, использование которых, может приводить к нарушению конфиденциальности, целостности, доступности системы и некорректной работе.

### Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Информационная система должна быть рассчитана на эксплуатацию в составе программно-технических средств Заказчика.

Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее обслуживание реализуется техническими и организационными средствами, предусмотренными в ИТ инфраструктуре Заказчика.

В процессе проведения периодического технического обслуживания ИС должны проводиться проверка параметров настроек работоспособности программных средств и тестирование их взаимодействия.

На основании результатов тестирования программных средств должны проводиться анализ причин возникновения обнаруженных дефектов и приниматься меры по их ликвидации.

Восстановление работоспособности системы должно проводиться в соответствии с инструкциями Разработчика и поставщика технических средств и документами по восстановлению работоспособности и завершаться проведением их тестирования. При вводе системы в опытную эксплуатацию должен быть разработан план выполнения резервного копирования программного обеспечения и обрабатываемой информации. Во время эксплуатации системы, персонал, ответственный за эксплуатацию системы должен выполнять разработанный план.

Все пользователи системы должны соблюдать правила эксплуатации электронной вычислительной техники. Клиенту необходимо предоставить всю необходимую документацию по пользованию, развертыванию системы и ее технической документации.

Квалификация персонала и его подготовка должны соответствовать технической документации.

### Требования по стандартизации и унификации

Разработка системы должна производится на базе особого архитектурного стиля взаимодействия компонентов распределенного приложения в сети, обеспечивающего унификацию, слабую связанность, модульность, повышение производительности, расширение и корректировку функционала в режиме промышленной эксплуатации без нарушения работы системы и простоту управления архитектурой и бизнес-логикой.

В качестве основного способа построения архитектуры АИС принять REST (Representational State Transfer — «передача состояния представления»),

В качестве прикладного протокола передачи данных принять HTTP.

Все элементы и компоненты системы, включая web-интерфейс, базу данных, должны использовать единую кодировку обмена, хранения и отображения данных в формате UTF-8.

В случае поступления информации от внешних источников данных в кодировке, отличающейся от UTF-8, АИС должна иметь функции приведения такой информации к унифицированному виду, пригодном для хранения в соответствие с требованием настоящего раздела.

В случае необходимости взаимодействия с компонентом, внешней системой, не подпадающей под требования настоящего раздела, в АИС должны использоваться общепринятые открытые протоколы, методы взаимодействия с внешними системами, системным и прикладным программным обеспечением.

### Архитектура

Должно быть предусмотрено сегментирование системы с выделением как минимум следующих сегментов: FrontEnd/DMZ, Middleware, BackEnd, DataBase.

Сегменты системы должны размещаться в различных подсетях (VU\N'ax). Должна быть разработана схема взаимодействия между сегментами системы, отражающая минимально необходимые наборы сетевых портов/протоколов.

Отказоустойчивость приложения должна обеспечиваться кластеризацией критически важных компонент. Работа кластеров должна по возможности обеспечиваться в режиме active-active.

Архитектура приложения должна предусматривать возможность «горизонтального» масштабирования путем увеличения количества компонент приложения (например, web и арр серверов) и балансировки нагрузки между ними в том числе за счет применения рекомендуемого к реализации механизма распределенной очереди задачи.

Вся программные компоненты должны иметь FOSS (Free and Open Source Software - бесплатное ПО с открытым исходным кодом) происхождение или разработанное лично исполнителем без нарушения чьих либо третьих лиц.

### Диагностика

При возникновении аварийных ситуаций, либо ошибок в программном обеспечении, диагностические инструменты должны позволять сохранять полный набор информации, необходимой Администратору или Разработчику для идентификации проблемы (логи, текущее состояние памяти, файловой системы).

### Перспективы развития, модернизации системы

При модернизации программного обеспечения должна сохраняться возможность внесения изменений без влияния на работу Системы в целом. Система должна поддерживать режимы обновления (патч-менеджмент) компонентов Системы и используемого ПО до актуальных версий программных средств.

### Дополнительные требования

Настоящее Техническое задание может быть уточнено в процессе проведения работ исходя из изменившихся требований Заказчика, которые могли быть не сформулированы на этапе разработки Технического задания.

Специальные требования по усмотрению Разработчика или Заказчика Системы оформляются соответствующими протоколами к настоящему Техническому заданию. Дополнительные работы не должны превышать 10% от общего объема работ. Отклонения от Технического задания необходимо согласовывать обеим сторонам.

# Требования к платформе

**Backend** системы должен:

* быть написан на языке последнего стабильного релиза, оптимизированном под большие нагрузки и работу; на усмотрение исполнителя, но с одобрением заказчика;
* иметь продуманный механизм развёртывания (Deployment) с использованием современной платформы или другого гибкого, быстрого и актуального механизма;
* быть написан по всем канонам современной разработки с использованием современных библиотек, с приоритетом на «открытый код»;
* учитывать, что максимальное критическое время на отображение конечного результата не превышает 1500 мс для обычных операций, таких как вывод информации и не более 5 секунд для сложных расчётов, таких как формирование отчётов;
* быть компонентным, расширяемым;
* быть масштабируемым (кластерной, учитывающая серверную виртуализацию).

### Требования к виртуализации:

* разделение на виртуальные машины
  + файловый сервер, сервер базы данных
  + web сервер, сервер приложений
* копирование данных
  + базы данных – ежедневно
  + файловые данные – еженедельно

### Требования к серверному обеспечению со стороны Заказчика

Заказчик предоставляет до передачи программного продукта оборудование следующей характеристики для настройки и установки системы исполнителем:

* Процессорная мощность не менее 16 поток
* Объем оперативной памяти не менее 16 Гига/байт
* Объем постоянной памяти не менее 1 Тера/байт
* 1 IP статичный адрес
* 1 интернет провайдеров с широкополосным доступом в интернет.
* Стабильное электропитание, обеспеченное запасным питанием

# Требования к документированию

Пакет эксплуатационной документации должен состоять из следующих документов и согласовываться с Заказчиком.

* Техническое задание на разработку системы;
* Структура баз данных;
* Руководство пользователя;
* Руководство системного администратора;
* План обучения;
* Программа и методика испытаний;
* Паспорт системы – документ с основными характеристиками системы, необходимыми в рамках ее эксплуатации, установки и развития;
* Инструкция проведения обновлений системы – руководство по развертыванию системы.

Вся разрабатываемая документация должна быть на русском языке и передаваться Заказчику в электронном виде.

# Ссылки

1. Чандрамохан D, Shibuya К, Р Setel, Кэрнкросс S, Lopez AD, Мюррей CJ, и др. Должны ли данные демографических систем наблюдения сделать более доступными для исследователей? PLoS Med 2008 Feb; 5 (2): E57
2. Byass P, Y, Берхане Emmelin A, D, Кебеде Andersson T, U, Hogberg и др. Роль демографических систем наблюдения (DSS) в оценке здоровья общин: пример из сельских районов Эфиопии. Public Health 2002 May; 116 (3): 145-150.
3. Baiden F, Ходжсон A, Binka FN. Демографические сайты наблюдения и возникающие проблемы в области международного здравоохранения. Bull World Health Organ2006 Mar; 84 (3): 163
4. Е. Y, Wamukoya M, Ezeh A, Эмина JB, Санкох O. Здоровье и демографические системы наблюдения: шаг в направлении полной регистрации актов гражданского состояния и системы статистики естественного движения населения в Африке к югу от Сахары Африке? BMC Public Health 2012 5 сентября, 12: 741
5. Сеть I. Население и здоровье в развивающихся странах, Том 1: Население, здоровье и выживание в INDEPTH Sites. Канада: МИЦР / CRDI; 2002.
6. Tanser Ж, Hosegood В, Bärnighausen Т, Herbst К, М Nyirenda, Muhwava Вт, и др. Когорта Профиль: Африка Центр демография информационная система (ACDIS) и обследование населения на ВИЧ. Int J Epidemiol 2008 Oct; 37 (5): 956-962
7. Кабан JJ, Гоц D. Визуальная аналитика в области здравоохранения - возможности и исследовательские задачи. J Am Med Assoc Inform 2015 Mar; 22 (2): 260-262.
8. AbouZahr C, информационные системы Боэрма Т. здравоохранения: основы общественного здравоохранения. Bull World Health Organ 2005 Август; 83 (8): 578-583
9. Кий D, G Adrienko, Фекете JD, Gorg С, Kolhammer Дж, Melancon G. Визуальные аналитики: определение, процесс, и вызовы. В: Карен А, Северо С, Stasko Дж, Фекете JD, редакторы. Информация Визуализация: Человеко-центрированный Проблемы и перспективы. Нью-Йорк: Springer; 2008.
10. Макнэбб CJ, Чунгонг S, Райан М, Wuhib Т, Р Нсубуга, Алему Вт, и др. Концептуальные основы надзора в области общественного здравоохранения и действия и его применение в реформировании сектора здравоохранения. BMC Public Health 2002; 2: 2
11. Карточка СК, Маккинли Дж, Шнейдерман В, редактора. Чтения в визуализации информации: Использование Зрения думать. Сан - Франциско, Калифорния, США: Morgan Kaufmann; 1999.
12. Ware C. Информация Визуализация: Восприятие Для дизайна (интерактивные технологии). Амстердам: Morgan Kaufmann;2012.
13. Ware C. Визуальное мышление: для проектирования (Morgan Kaufmann Series в интерактивных технологий). Амстердам: Morgan Kaufmann; 2008.
14. Аль-Хадж S, Pike I, Рикк B, Fisher B. Визуальные аналитики для общественного здравоохранения: Поддержка знаний строительства и принятие решений. 2013 Представлено на: 2013 46-я Гавайи Международной конференции по системе наук; 7-10 января, 2013; Wailea, Мауи, Гавайи, США р. 2416-2423.
15. Лю Z, Stasko JT. Ментальные модели, визуальное мышление и взаимодействие в визуализации информации: сверху вниз перспектива. IEEE Trans Вис вычи График 2010; 16 (6): 999-1008.
16. Миллер Г.А.. Магическое число семь плюс или минус два: некоторые ограничения на нашу способность к обработке информации. Psychol Rev 1956 Mar; 63 (2): 81-97.
17. Tegarden DP. Бизнес-визуализация информации. Commun Assoc Inf Syst 1999; 1: 4.
18. Лавалья S, Малы Е, Р Шок, Хопкинс М.С., Kruschwitz Н. MIT Sloan Обзор управления. 2011. Больших данных, аналитика и путь от идеи до значения
19. Ченг СК, Ip ДК, Коулинг BJ, Хо Л.М., Leung Г.М., Лау EH. Цифровой дизайн приборной панели с использованием нескольких потоков данных эпиднадзора за болезнями с эпиднадзора за гриппом в качестве примера. J Med Res Интернет 14 октября 2011; 13 (4): E85
20. Немного S. Панель информации Дизайн: Эффективная Визуальная передача данных. Сан - Франциско: O'Reilly Media;2006.
21. Векслер S, Шаффер C, Cotgreave А. Большая книга щитков: визуализация данных с использованием реальных - Всемирный бизнес
22. Сарикая A, Correll M, L Bartram, тори M, Fisher D. Что мы говорим о том, когда мы говорим о приборных панелях? IEEE Trans Вис вычи График 2019; 29 (1): 682-692.
23. Кэрролл Л., Au AP, Detwiler LT, Fu TC, Painter IS, Abernethy NF. Визуализация и аналитика инструменты для эпидемиологии инфекционных заболеваний: систематический обзор. J Biomed Inform 2014 Oct; 51: 287-298
24. Sopan A, Нох A, S Karol, Rosenfeld P, Ли G, Шнейдерман B. Сообщество Карта здравоохранения: геопространственный и многомерный инструмент визуализации данных для наборов данных в области общественного здравоохранения. Г Inf В 2012 Apr; 29 (2): 223-234.
25. Kostkova P, S Garbin, Moser J, Pan W. интеграции и визуализации панели общественного здравоохранения: МЕДИ + доска пилотный проект. В: Труды 23 Международной конференции по World Wide Web. Нью-Йорк: ACM; 2014 Представлено на: WWW'14 Companion; Апрель 7-11, 2014; Сеул, Корея р.657-662.
26. Шнейдерман Б. Глаза есть это: задание по типу данных таксономии для информационных визуализаций. В: The Craft визуализации информации. Амстердам: Эльзевир; 2003.
27. Chen C, исследование Ю. Ю. Эмпирические визуализации информации: мета-анализ. Int J Hum вычи стад 2000 ноябрь; 53 (5): 851-866.
28. Северная C. К измерению визуализации проницательности. IEEE вычи Grap Appl 2006 May; 26 (3): 6-9.
29. Амар R, Stasko J. Task Based-рамочная структура знаний для проектирования и оценки информации Визуализации. В: Труды IEEE симпозиума по визуализации информации. 2004 Представлено на: INFOVIS'04; 10-12 октября, 2004; Austin, TX р. 143-150.